



Unidad #2

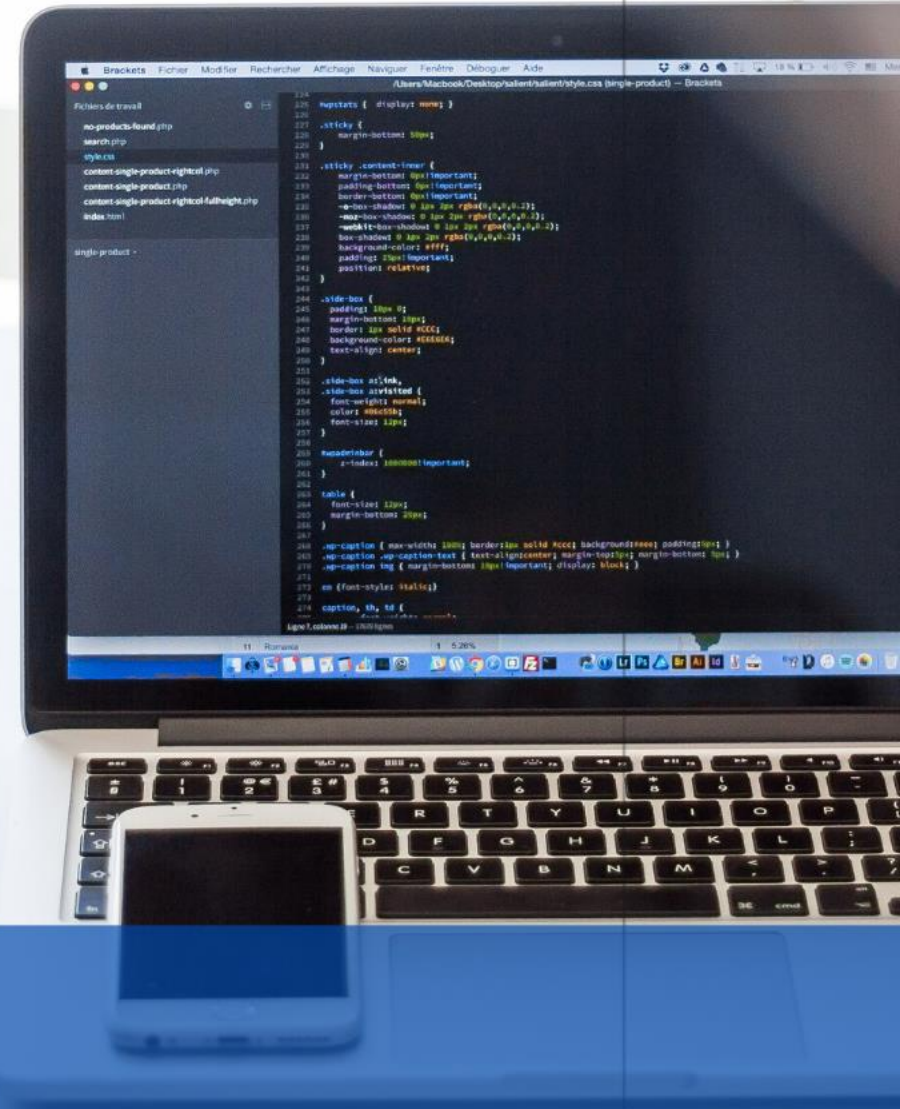
Arrays.

FUNDAMENTOS DE
PROGRAMACION

Ing. Timotea Guadalupe Menjivar

Fundamentos de Programación

ING. TIMOTEA GUADALUPE MENJIVAR



Introducción al contenido

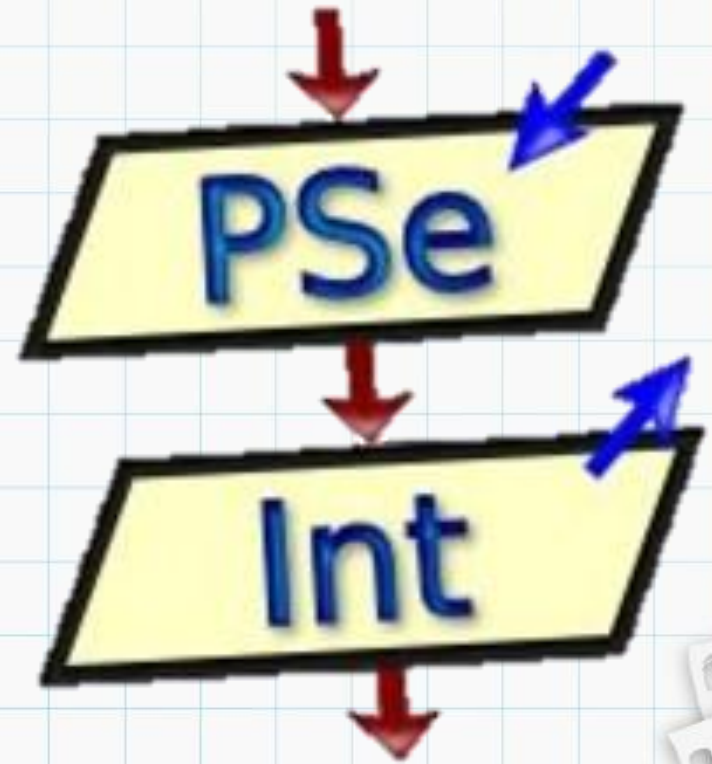
En los contenidos anteriores se han abordado los elementos fundamentales que nos permitirán avanzar en el mundo de la programación, con estos contenidos en mente llegamos a los fundamentos de programación: las estructuras extendidas de la programación. Indistintamente del lenguaje que se utilice, los arreglos pueden pensarse como vectores, matrices, etc.



A man in a suit and headset is working on a laptop. The image is overlaid with a blue gradient. The text "Desarrollo del contenido" is centered in white.

Desarrollo del contenido

Herramienta de Pseudocódigo



¿Qué es una matriz(arreglos, arrays)?

Los arreglos son estructura de datos homogéneas (todos los datos son del mismo tipo) que permiten almacenar un determinado número de datos bajo un mismo identificador, para luego referirse a los mismos utilizando uno o más subíndices.



La norma común en toda declaración de arreglos reside en que la variable a la que se va a asignar el arreglo debe ir seguida de los signos de paréntesis, indiquemos o no el número elementos.



El inicio del array es de 0

5 elementos
En el array

Índice	Valor
0	
1	
2	
3	
4	

→ El valor que se almacena
en el array



Para crear un arreglo en PSeInt se utiliza la palabra clave Dimension, seguido del nombre del arreglo (identificador) y su tamaño (numero de subíndices) entre corchetes ().

Nota: antes de crear la dimensión se debe definir la variable.

```
2      definir a como entero;  
3      dimension a(3);
```


Como ejemplo crearemos un programa que permita ingresar tres números guardarlos en un array e imprimirlos en orden inverso. Lo primero que haremos es definir las variables y la dimensión(array):

<sin_titulo>* X

```
1  Proceso numeros
2      // definimos la dimension a y la variable para el ciclo b
3      definir a,b como entero;
4      //creamos la dimension como entera
5      dimension a(3);
6      //iniciamos la variable para el ciclo y posicion del array
7      b<-1;
8      "
```

Creamos la dimensión y con un bucle mientras guardamos los datos.

Proceso numeros

```
// definimos la dimension a y la variable para el ciclo b
definir a,b como entero;
//creamos la dimension como entera
dimension a(3);
//iniciamos la variable para el ciclo y posicion del array
b<-1;
//creamos un ciclo mientras
Mientras b<3 Hacer
    Escribir "ingrese un numero: ";
    //mandamos a guardar el numero en el array
    Leer a(b);
    b<-b+1;
FinMientras
```

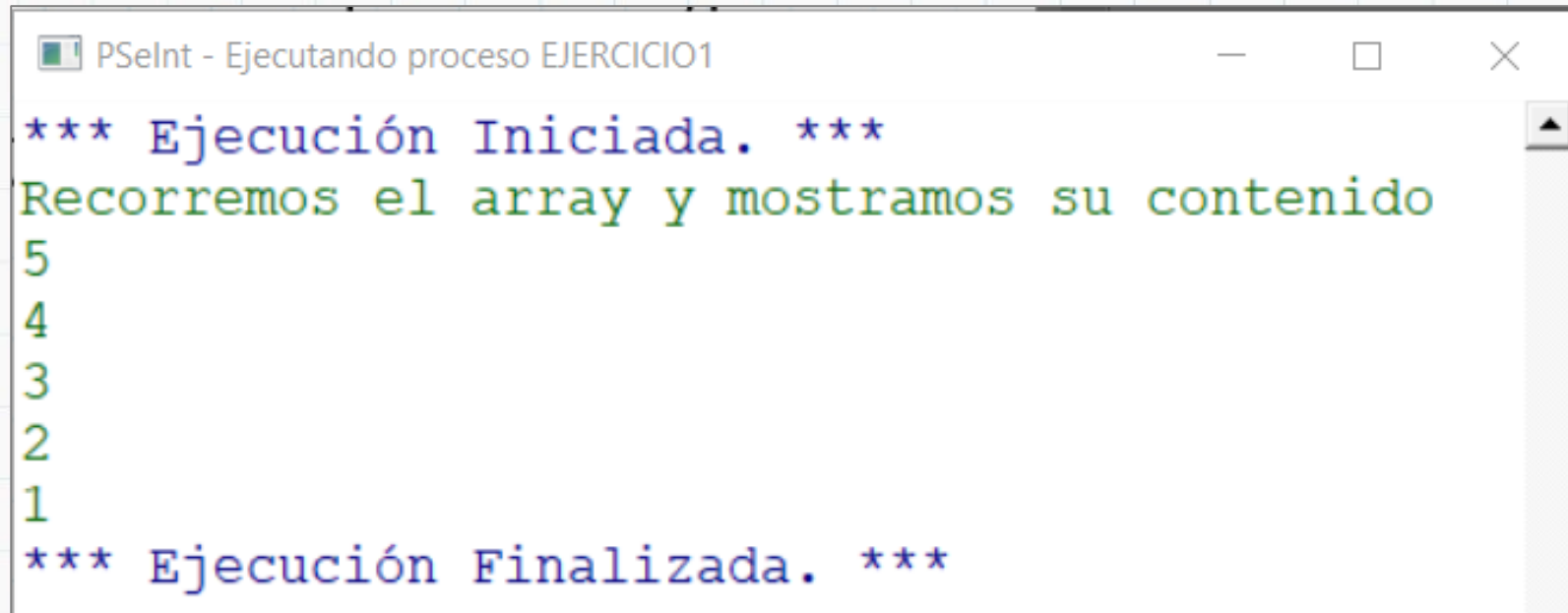
Ahora solo queda que nos genere los números de forma inversa.

```
1  Proceso numeros
2      // definimos la dimension a y la variable para el ciclo b
3      definir a,b como entero;
4      //creamos la dimension como entera
5      dimension a(3);
6      //iniciamos la variable para el ciclo y posicion del array
7      b<-1;
8      //creamos un ciclo mientras
9      Mientras b<3 Hacer
10         Escribir "ingrese un numero: ";
11         //mandamos a guardar el numero en el array
12         Leer a(b);
13         b<-b+1;
14     FinMientras
15     Para b<-2 Hasta 1 Con Paso -1 Hacer
16         escribir a(b);
17     FinPara
18 FinProceso
```

Crea un array o arreglo unidimensional con un tamaño de 5, asigne los valores numéricos manualmente (los que tu quieras) y muéstralos por pantalla.

```
1  Proceso Ejercicio1
2      //Declaramos variables
3      Definir num,i como entero;
4      Dimension num(5);
5      //Asignamos valores
6      num[0]<-1;
7      num[1]<-2;
8      num[2]<-3;
9      num[3]<-4;
10     num[4]<-5;
11     //Recorremos el array y mostramos su contenido
12     Escribir "Recorremos el array y mostramos su contenido";
13     Para i<-4 Hasta 0 Con Paso -1 Hacer
14         escribir num[i];
15     FinPara
16
17 FinProceso
```


Crea un array o arreglo unidimensional con un tamaño de 5, asígnale los valores numéricos manualmente (los que tu quieras) y muéstralos por pantalla.

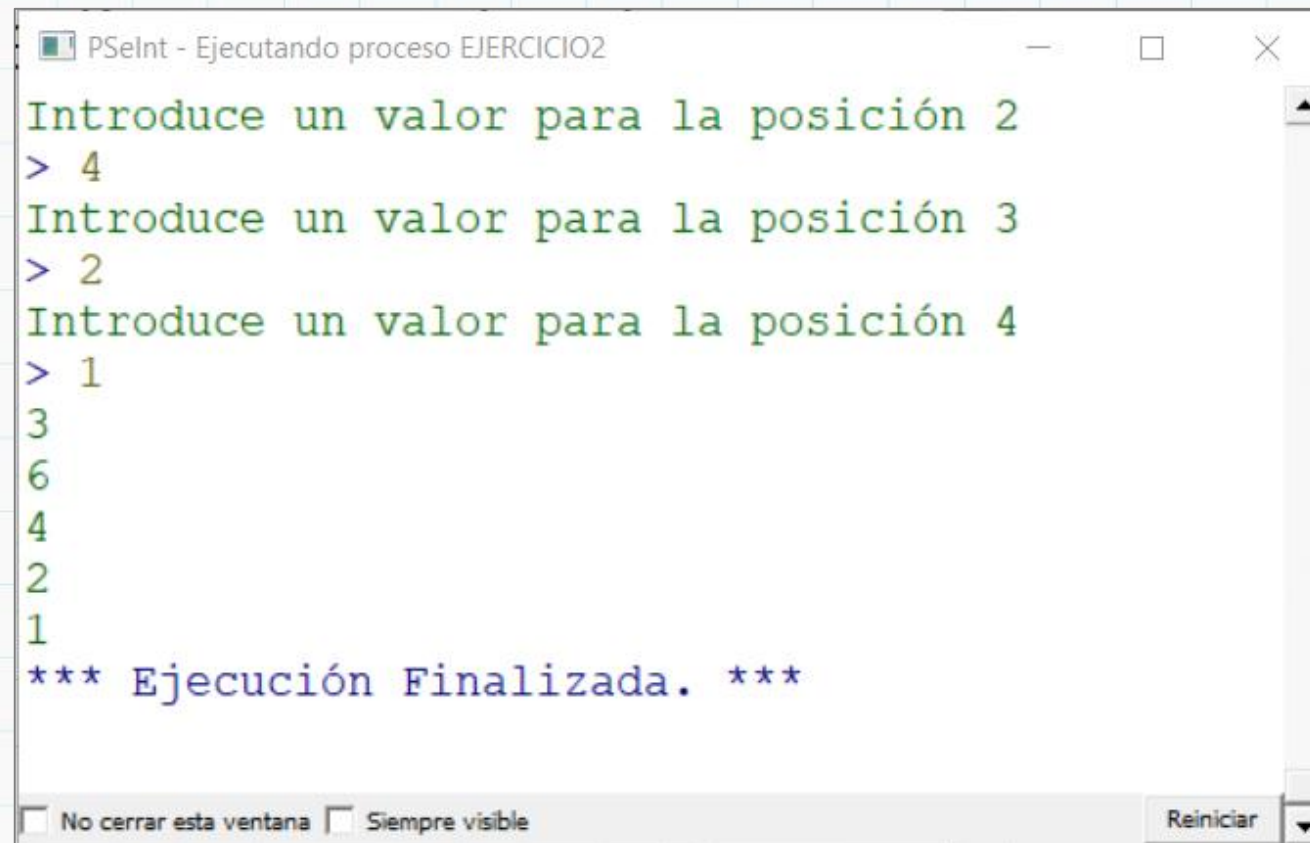


```
*** Ejecución Iniciada. ***
Recorreremos el array y mostramos su contenido
5
4
3
2
1
*** Ejecución Finalizada. ***
```

Modifica el ejercicio anterior para que insertes los valores numéricos con **leer** mediante un bucle y los muestre por pantalla.

```
1  Proceso Ejercicio2
2      //Declaramos variables
3      Definir num,i como entero;
4      //Definimos el array
5      Dimension num(5);
6
7      //Introducimos valores
8      Para i<-0 Hasta 4 Con Paso 1 Hacer
9          Escribir "Introduce un valor para la posición ",i;
10         leer num(i);
11     FinPara
12     //Recorremos el array y mostramos su contenido
13     Para i<-0 Hasta 4 Con Paso 1 Hacer
14         escribir num(i);
15     FinPara
16 FinProceso
```

Modifica el ejercicio anterior para que insertes los valores numéricos con **leer** mediante un bucle y los muestre por pantalla.

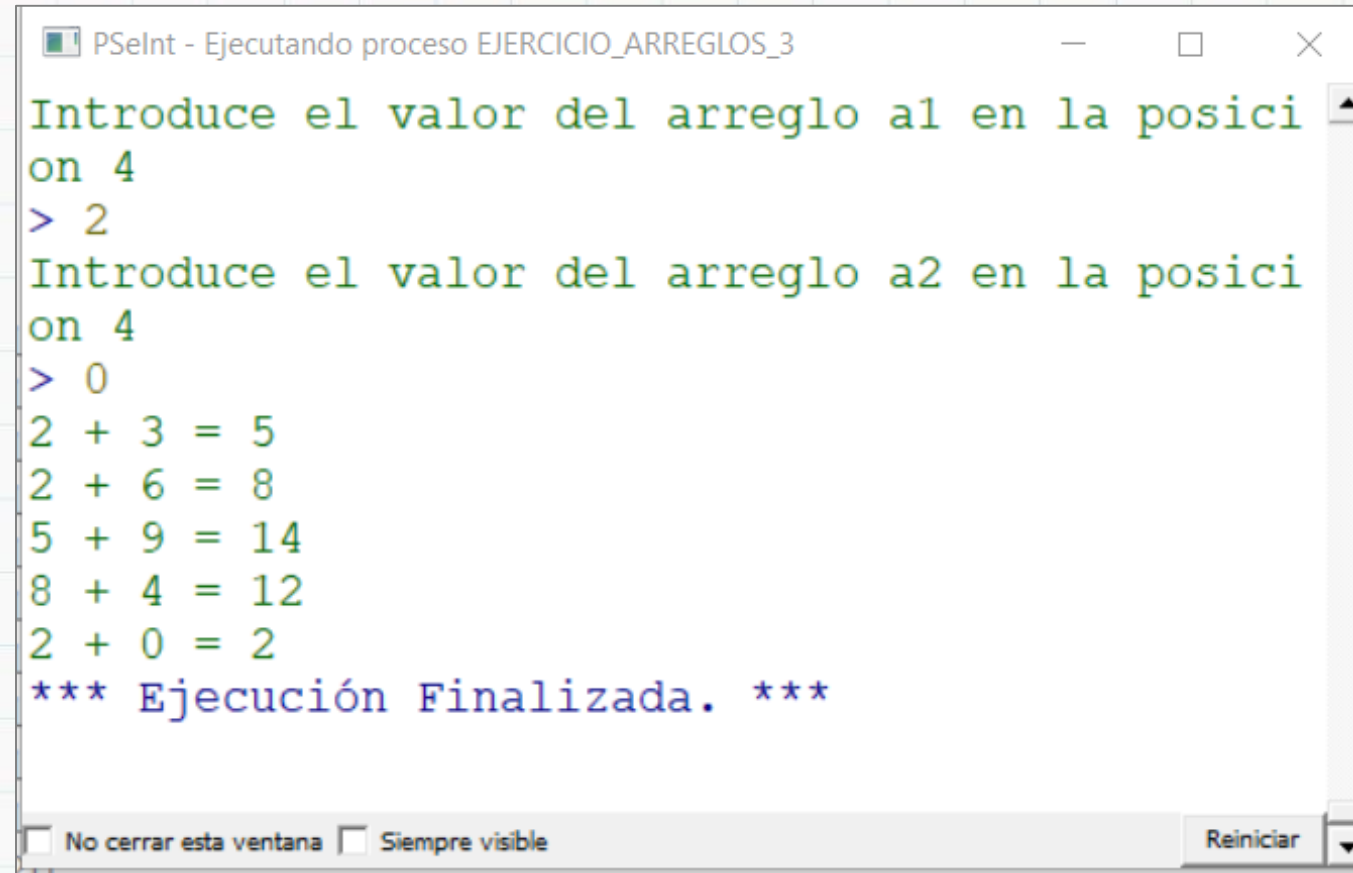


```
PSeInt - Ejecutando proceso EJERCICIO2
Introduce un valor para la posición 2
> 4
Introduce un valor para la posición 3
> 2
Introduce un valor para la posición 4
> 1
3
6
4
2
1
*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

Pedir valores numéricos en dos arrays distintos y almacenar el resultado de los valores de cada posición (posición 0 del arreglo 1 + posición 0 del arreglo 2) y mostrar el contenido de los 3 arreglos de esta forma. valor pos 0 arreglo 1 + valor pos 0 arreglo 2 = valor pos 0 arreglo 3 valor pos 1 arreglo 1 + valor pos 1 arreglo 2 = valor pos 1 arreglo 3 ...

```
1  Proceso ejercicio_arreglos_3
2      definir al,a2,a3,i como entero;
3      Dimension al(5);
4      Dimension a2(5);
5      Dimension a3(5);
6      //Recorremos de 0 a Tamaño-1
7      //Podemos rellenar el arreglo al, a2 y a3
8      Para i<-0 Hasta 5-1 Con Paso 1 hacer
9          Escribir "Introduce el valor del arreglo al en la posicion ",i;
10         leer al(i);
11         Escribir "Introduce el valor del arreglo a2 en la posicion ",i;
12         leer a2(i);
13         //en el arreglo a3 guardamos la suma de los otros dos.
14         a3(i)<- al(i) + a2(i);
15     FinPara
16     //Mostramos los datos en el formato dicho
17     Para i<-0 Hasta 5-1 Con Paso 1 hacer
18         Escribir al(i) , " + ",a2(i), " = ", a3(i);
19     FinPara
20
21 FinProceso
```


Pedir valores numéricos en dos arrays distintos y almacenar el resultado de los valores de cada posición (posición 0 del arreglo 1 + posición 0 del arreglo 2) y mostrar el contenido de los 3 arreglos de esta forma. valor pos 0 arreglo 1 + valor pos 0 arreglo 2 = valor pos 0 arreglo 3 valor pos 1 arreglo 1 + valor pos 1 arreglo 2 = valor pos 1 arreglo 3 ...



```
PSeInt - Ejecutando proceso EJERCICIO_ARREGLOS_3

Introduce el valor del arreglo a1 en la posición 4
> 2
Introduce el valor del arreglo a2 en la posición 4
> 0
2 + 3 = 5
2 + 6 = 8
5 + 9 = 14
8 + 4 = 12
2 + 0 = 2
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

Guia de Ejercicios

- Realice un programa en P-seint que genere los números naturales desde la unidad hasta el numero que escriba el usuario. Luego que nos muestre cuales son múltiplos de 2, múltiplos de 3 múltiplos de 5.
- Realice un programa en P-seint que genere números del 1 al 100 y sume solo aquellos números que son impares.
- Realice un programa en P-seint que solicite como entrada dos números por el teclado, imprima todos los números que están entre ese rango, que cuente cuantos números están en total en ese rango y cuantos de ellos son pares.
- Realice un programa que cuente el numero de veces que se repite un determinado numero.

Guia de Ejercicios

- Cree un programa que genere los números del 1 al 20 pero que solo almacene en un array o dimensión los números pares.
- Cree un programa que guarde en un array o dimensión los nombres de 5 personas
- Cree un programa que guarde en un array o dimensión los nombres de los meses del año.

Recursos Complementarios

Recurso	Título	Cita Referencial
Libro	<i>Fundamentos de programación 4ta Edición</i>	Aguilar, L. J. (2008). <i>Fundamentos de programación 4ta Edición</i> . Madrid: McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A. U.

Actividad de la semana

Nombre de la actividad	Creando Ejercicios
Tipo de actividad	Foro
Tipo de participación	Colaborativa (individual)
Instrucciones para la actividad	<p>Para el desarrollo de esta actividad el alumno deberá elaborar 1 ejercicio tipo desafío(no debe publicarse la solución solo el enunciado) para realizarlo utilizando las estructuras repetitivas y vectores. El ejercicio no puede repetirse entre estudiantes. Por ello deben ser cuidadosos al publicar y verificar si antes no se ha publicado ese ejercicio por otro participante.</p> <p>El participante deberá compartir los ejercicios en el foro habilitado para la actividad: Foro Creando Ejercicios</p>
Fecha de entrega	La fecha límite de participación será el día domingo 10 de mayo a las 11:55 p.m.

Nombre de la actividad	Herramientas para Servidores Web
Tipo de actividad	Tarea
Tipo de participación	Colaborativa (3 integrantes)
Instrucciones para la actividad	<p>Después de haber dado lectura a los contenidos de esta semana, ingrese a: <i>Tarea Actividad de la semana del 11 al 17 de abril-Resolucion de Guia</i>, en la cual tomando ejemplo de los contenidos desarrollados en clase deberá:</p> <ul style="list-style-type: none"> • Desarrollar la guia en equipos de 3 integrantes • Solucionar los ejercicios propuestos en el material de la semana. • Cree una caratula con los integrantes del equipo. • Solucione los ejercicios desarrollados en pseint. • Con la información recabada subir los ejercicios en una carpeta comprimida junto con la caratula.
Fecha de entrega	La fecha límite de participación será el día domingo 17 de mayo a las 11: 55 p.m.
Criterios de evaluación	<ul style="list-style-type: none"> • Seguimiento de instrucciones (2.0 puntos) • Uso correcto de las estructuras (3.0 puntos) • Solucion de la guia de trabajo (5.0 puntos)